Swedish Language Processing in the Spoken Language Translator

Björn Gambäck
Natural Language Processing Group
Swedish Institute of Computer Science
Box 1263, S - 164 28 KISTA
Stockholm, Sweden
gamback@sics.se

Abstract

The paper describes the Swedish language components used in the Spoken Language Translator (SLT) system. SLT is a multi-component system for translation of spoken English into spoken Swedish. The language processing parts of the system are the English Core Language Engine (CLE) and its Swedish counterpart, the S-CLE. The S-CLE is a general purpose natural language processing systems for Swedish which in the SLT project was tuned towards the register of the air travel information (ATIS) domain. The peculiarities and the coverage of the resulting Swedish grammar are the main topics of the paper, even though the overall SLT system also is briefly described.

1 Introduction

The Swedish Core Language Engine (or S-CLE for short) (Gambäck and Rayner, 1992) is a general purpose natural language processing system for Swedish developed by the Swedish Institute of Computer Science from its English counterpart, the SRI Core Language Engine (CLE) (Alshawi, 1992). The key idea behind the system is indicated by the word "core": the S-CLE was intended to be used as a building block in a broad range of applications and has already been tested as part of a database query system (Gambäck and Ljung, 1993) and as a text-to-speech frontend (Gambäck and Eineborg, 1995). The two copies of the CLE have also been used together to form a machine translation system for a car-hire domain (Alshawi *et al.*, 1991).

In the Spoken Language Translator, described in the next section, the English CLE performed as a back-end to a speech recognition system, the S-CLE as a frontend to a speech synthesis system, and the two CLEs together formed a (text) translation system in the air travel information domain. In the course of the project, the previous Swedish system was completely redesigned and the general-purpose grammar expanded, but also tuned to cover the peculiarities of the register (sublanguage) of a particular domain.

Proceedings of NODALIDA 1995, pages 37-49

The present paper starts out by describing the overall SLT system architecture in Section 2 and briefly introduces the different components of the system. Section 3 is the main focus of the paper, describing the different modules of the present Swedish language processing component in detail by giving examples of the rules used for the treatment of some specific phenomena.

Section 4 details the coverage issues and how the Swedish coverage was improved during the first year of the project. The final section of the paper looks into the future, describes the ongoing work on making the system completely bidirectional, and sums up the previous discussion.

2 The SLT system

The Spoken Language Translator (SLT) is a system prototype which can translate queries from spoken English to spoken Swedish in the domain of air travel planning (ATIS). The system was developed as a joint effort by the Swedish Institute of Computer Science, SRI International (Menlo Park, US and Cambridge, UK), and Telia Research AB (Haninge, Sweden). Most of the first-year prototype was constructed from previously existing pieces of software, which were adapted for use in the speech translation task with as few changes as possible. The overall architecture of the current version of SLT system is described shortly in this section, for a complete description see (Rayner *et al.*, 1993) or (Agnäs *et al.*, 1994).

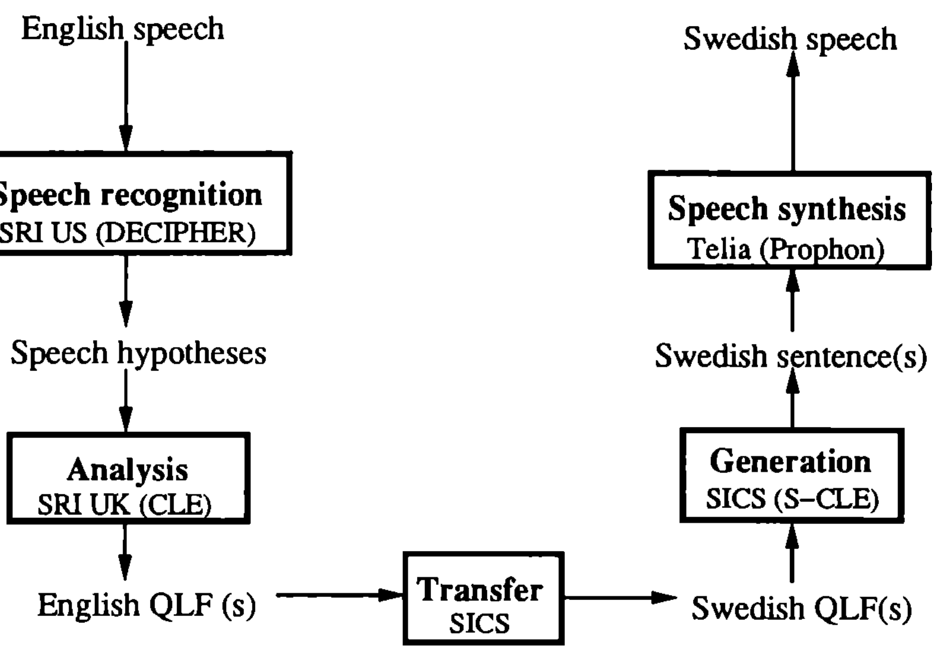

Figure 1: Top-level architecture of the Spoken Language Translator

The main components of the SLT system are connected together in a pipelined sequence as shown in Figure 1. The input signal is processed by SRI Menlo Park's DECIPHER(TM) (Murveit *et al.*, 1991), a speaker-independent continuous speech recognition system based on Hidden Markov Model technology. It produces a set of speech hypotheses which is passed to the English-language processor, the SRI Cambridge Core Language Engine (Alshawi, 1992).

The CLE grammar associates each speech hypothesis with a set of possible quasi-logical forms, QLFs (Alshawi and van Eijck, 1989), typically producing 5 to 50 QLFs per hypothesis. In order to allow fast processing of a large number of hypotheses, a scaled-down version of the grammar induced with the machine-learning technique "Explanation-Based Learning" (Samuelsson and Rayner, 1991) is first invoked and parsed with an LR-parser (Samuelsson, 1994). Only if this restricted-coverage grammar fails is the general-purpose grammar tried on the (by the speech recognizer) most preferred hypothesis.

A preference component is then used to give each QLF a numerical score reflecting its linguistic plausibility (Alshawi and Carter, 1994). When the preference component has made its choice, the highest-scoring logical form is passed to the transfer component, which uses a set of simple non-deterministic recursive pattern-matching rules to rewrite it into a set of possible corresponding Swedish representations (Alshawi *et al.*, 1991; Gambäck and Bretan, 1994).

The preference component is now invoked again, to select the most plausible transferred logical form. The result is fed to a second copy of the CLE, which uses a Swedish-language grammar and lexicon developed at SICS (Gambäck and Rayner, 1992) to convert the form into a Swedish string and an associated syntax tree. Finally, the string and tree are passed to the Telia Prophon speech synthesizer, which utilizes polyphone synthesis to produce the spoken Swedish utterance (Bäckström *et al.*, 1989).

The SLT system's current performance figures measured on previously unseen data (the 1001-utterance December 1993 ATIS corpus) are: 78.8% of all utterances are such that the top-scoring speech hypothesis is an acceptable one. If the speech hypothesis is correct, then an acceptable translation is produced in 68.3% of the cases and the overall performance of the system is 53.8%. Limiting the test corpus to sentences of 10 words or less (688 utterances), these figures move up to 83.9% for speech recognition and 74.2% for language processing, with a 62.2% overall performance.

For about 10% of the correctly recognized utterances, an unacceptable translation is produced. Nearly all of these are incorrect due to their containing errors in grammar or naturalness of expression, with errors due to divergence in meaning between the source and target sentences accounting for less than 1% of all translations. SLT performance is discussed at length in (Rayner *et al.*, 1994).

3 Swedish Language Processing

As noted above, the S-CLE is a general purpose natural language processing system for Swedish. The main object of the system is to map certain natural language expressions into appropriate predicates in quasi-logical form. The system is based completely on unification and has a fairly large bidirectional phrase-structure type grammar (i.e., the grammar can be used both for analysis and generation) covering most of the common Swedish constructions. There is a good treatment of inflectional morphology, covering all main inflectional classes of nouns, verbs and adjectives.

The S-CLE has been developed from the original English CLE by replacing English-specific modules (grammar, morphology, lexicon and lexicon acquisition) with corresponding Swedish-language versions, exploiting the large overlap between the structures of the two languages. Most of the Swedish grammar is thus completely equivalent to the English one; this section will concentrate on the parts that differ for interesting reasons. (So, even though the grammars indeed differ in several ways not described here, most of the differences are for rather uninteresting reasons more reflecting different tastes on the side of the grammarians than real grammatical differences and will thus be left out from the discussion here.)

A previous version of the Swedish grammar and how it was developed was described in (Gambäck and Rayner, 1992). There we also went into some detail on the (at least for a translation task) most vital differences between English and Swedish, both at the morphology and syntax levels. The present paper will thus refrain from recapitulating that discussion and only give an overview of the most important phenomena and their present treatment in the system.

First, however, we should note that the simple methodology outlined for developing a system for a new language has also been shown to be successful for other languages. A full-scale version of the CLE for French has recently been developed by ISSCO, Geneva. It has a coverage at roughly the same level as the Swedish one (Rayner and Bouillon, 1995) and is also used as a part of a spoken language translation system (English-to-French), which was demonstrated at the CeBIT fair in Hannover, March 1995.

Small-scale versions of the CLE are also under development by the University of Cambridge: for German (Parkinson, 1992), mainly for testing the grammar formalism on a language with a different word order; for Polish (Styś, 1995), testing the morphology component on the intricacies having to do with case, gender and number variation on nouns, as well as the noun phrase part of the grammar on some of the problems associated with a "free" word order; and for Korean.

The rest of this section will in turn go through the different processing steps used when forming a QLF in the S-CLE and describe the rule sets used in each of them: first the morphological processing where the rulebase is divided into morphophonological "spelling" rules and morphosyntactic "production" rules. Then the grammatical processing which in turn is divided into two steps, syntactic parsing and semantic analysis. The rules of the grammar proper are thus divided into two different rule sets, one with the syntax and another with the (compositional) semantics. The main processing chain is as shown in Figure 2.

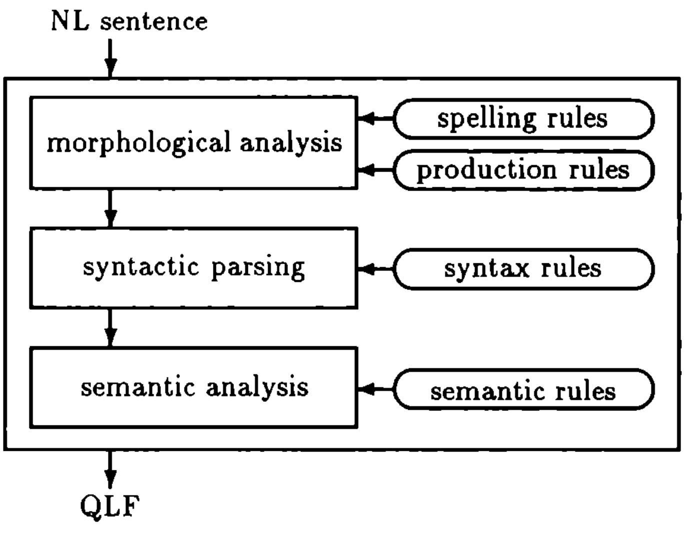

Figure 2: The analysis steps of the S-CLE

3.1 Morphology

Given that Swedish is an inflectional language, the treatment of the inflectional morphology by simple affix-stripping used in the original English CLE was far from sufficient. A "lazy" version of the two-level morphology (Koskenniemi, 1983) was thus implemented (Carter, 1995). This version is "lazy" in that it does not account for general changes of the stems of words.

A typical spelling rule is the following which shows that when the affix `er` is added to a stem ending with an `o` or an `e` followed by an `l` or an `r`, the stem vowel may be dropped unless it is stressed (i.e., formler = formel + er, manövrer = manöver + er, etc.):

```
spell(plur_LRer_eLR,
      "||", =>, "|2|1+er",
      [2/"oe",1/"lr"],
      [stresslast=n]).
```

In all the rule formalisms of the CLE, the first argument (here, `plur_LRer_eLR`) is simply a rule name mainly used for debugging. The main parts of the rule appear on the different sides of the arrow (`=>`): these are the surface and lexical forms, respectively. The vertical bars (`|`) indicate which letters may be changed in the rule. If the arrow is bidirectional (`<=>`), the rule *must* apply; here it may optionally apply. The final two lists put restrictions on the "variables" `1` and `2` in the rule, and on possible feature settings on the stem.

In the current version of the Swedish morphology (which is still under development), 58 such spelling rules appear and are complemented by another set of 4 interword rules used in the derivational morphology, which in Swedish is also quite complex; however, since the current version of the system cannot handle derivational morphology in general, we will not go into too much detail here, but concentrate on the — for the task at hand — most important part of it, namely the production of noun compounds, which are extremely common in the ATIS domain.

While noun compounds in English are formed simply as groups of words, the Swedish compounds are formed by actually compounding the words together. In general, this can be done in a wide variety of fashions, but in present-day Swedish mainly in two ways only: either by just "gluing" the words together, or by inserting an *-s-* between the words in the compound, as described in for example (Kiefer, 1970). Compounds can in general be of almost all word-classes, but the most common ones are noun compounds, in which the last word of the compound is a noun; the other words in the compound can be of other classes (e.g., adjectives or adverbs), but are normally nouns, as well.

As a rule-of-thumb, noun compounds are formed first without inserting an infix *s*, but if the compound consists of more than two words, an *s* will be inserted for every second word added to the compound, so for example the following sequence would give the words for "father", "grand-father" (father's father), "great grand-father", etc.:

far, *farfar*, *farfarsfar*, *farfarsfarfar*, ...

Whether a particular noun will form compounds by inserting an *s* or not depends on the word in question and is thus lexicalized.

To implement the noun compound formation, a feature **nn_infix** on an $\overline{\text{N}}$ (nouns are lexicalized as $\overline{\text{N}}$s) indicates whether or not it can be post-modified with a complex $\overline{\text{N}}$ compound. The feature can currently take the values **s** (for an infix **'-s-'**) or **n** (for no infix, in practice **'--'**, i.e., just a hyphen) and is lexicalized on the $\overline{\text{N}}$, thus indicating the fact that some lexicon $\overline{\text{N}}$s take an s-infix when forming a compound, and some do not.

The following morphological production rule for noun-noun compounds (there are similar rules for other types of noun compounds) show together with the two rules for infixes (**'-s-'** or **'--'**) how the **nn_infix** feature propagates as 00+01=01, 01+10=00, i.e., an $\overline{\text{N}}$ that takes no infix has **nn_infix(0,0)** as its lexicon value and meets the **'--'**-infix which has **nn_infix(0,1)** to produce an $\overline{\text{N}}$ with **nn_infix(0,1)**, which in turn can produce an $\overline{\text{N}}$ that takes the null-infix if it meets the **'-s-'**-infix (**nn_infix(1,0)**), etc.

```
  nbar:[nn_infix=(I,0), simple=n, ...]
   -->
  nbar:[nn_infix=(I,N), ...]
   +
  'INFIX':[nn_infix=(N,0)]
   +
  nbar:[simple=y, ...]

lex('-s-',['INFIX':[nn_infix=(1,0)]]).
lex('--',['INFIX':[nn_infix=(0,1)]]).
```

The setting of the feature **simple** force complex compounds to form in a left-branching fashion; the right-most daughter may not itself be a compound (must have **simple=y**).

Production rules like the one above currently number 27 in the system, only 4 of which are used for forming compounds. These production rules are actually used by the syntactic morphological processing and are more or less paralleled by 33 semantic morphological derivation rules.

3.2 Syntax

On the syntactic side, the English and Swedish grammars differ on many accounts. Firstly, several extra rules appear in the Swedish system, mainly to capture different kinds of movements, in particular the fact that Swedish allows for topicalization of just about any type of constituent. Space considerations prevent a full account of these rules from being included in this paper; they will be reported on elsewhere (Gambäck, 1995). Here, we will thus only concentrate on some prototypical cases.

Secondly, a number of new features had to be added or the value ranges or relevant rules for old features had to be extended. Most notably since the more complex agreement structure of Swedish means that the features indicating agreement and definiteness must be propagated to many more constituents.

An example is the three-valued definiteness feature, which ranges over values for "indefinite", "definite" and "possessive", the last one being used on genitive NPs. These are treated as forming complex determiners, so that '*en mans fru*' (a man's wife), '*mannens fru*' (the man's wife), and '*Kalles fru*' (Kalle's wife) are all interpreted as having the structure [NP [DET $\overline{\text{N}}$]] as examplified in Figure 3.

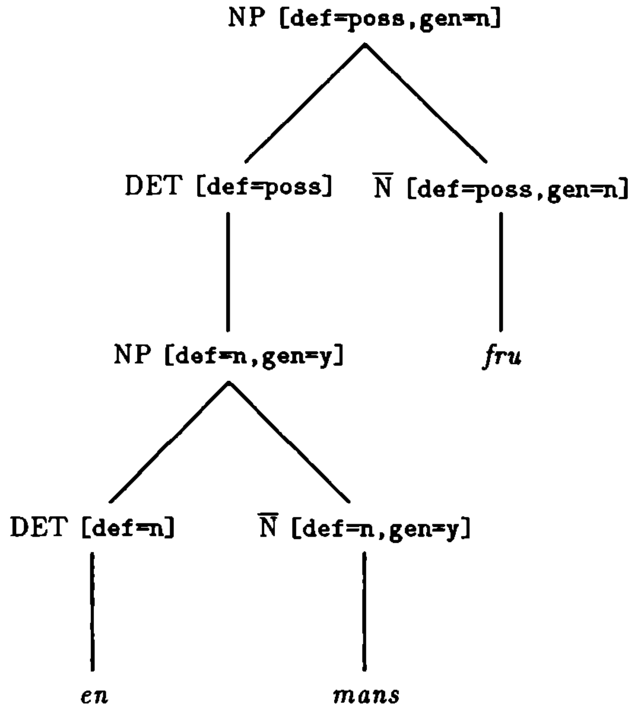

Figure 3: The tree structure for the noun phrase '*en mans fru*'

This is obtained by using the following two rules (here quite simplified with most features removed):

```
syn(det_np_Genitive,
[det:[def=poss],
  np:[def=_, gen=y]
]).

syn(np_det_nbar,
[np:[def=D, gen=G],
  det:[def=D],
  nbar:[def=D, gen=G],
]).
```

The first rule specifically forms determiners from genitive NPs (with the feature setting gen=y) regardless of the NP's definiteness (def=_), giving the newly formed determiner a possessive definiteness. The second rule forms NPs from determiners and nouns as long as the definiteness values on the daughters unify. This rule may be used on a wide range of determiner and noun types, including genitives.

3.3 Semantics

Most of the differences between English and Swedish syntax is only mirrored at the (QLF, i.e., compositional) semantic level without any interesting additions. The most notable exception is the verb-phrases. Already at the syntax-level, most word-order differences stem from the strongly verb-second nature of Swedish: formation of both YN- and WH-questions is by simple inversion of the subject and verb, without the introduction of an auxiliary. This is illustrated in the following examples:

Han såg Maria.	*He saw Mary.*
Såg han Maria?	*Did he see Mary?*
Vem såg han?	*Who did he see?*

This difference of verb syntax can actually be factored away. However, we will not dwell in too much detail on the rather special unification-based treatment of verb-phrases used in the system — for that, the reader is referred to (Gambäck, 1993a; Gambäck, 1993b) — but will note that the main trick used is *lexicalization*: information regarding for example verb subcategorization schemes (i.e., the number and type of verbal complements, such as objects, particles, etc.) is removed from the grammar and put in the lexicon instead. Syntactically, this enables us to treat both English and Swedish verb-phrases of different kinds with a rule like the following:

```
syn(vp_v_comp_Normal,
[vp:[tense_aspect=TA],
  v:[aux=_, tense_aspect=TA,
     subcat=Complements]
  | Complements
]).
```

where the value of the `subcat` feature of the verb has to unify with the rest of the verb-phrase. The value of `subcat` is specified for a particular verb in its lexical entry and can of course be empty (for intransitives, etc.). Our current Swedish grammar treats 48 different main verb complement patterns plus copulas and auxiliaries. Without claiming this to be the absolute number of Swedish verb types in any sense, it is easily understandable that without the strategy outlined above, we would have been forced to state specific instances of the verb-phrase formation rule for a vast number of cases.

In the CLE, each syntactic rule is paralleled by (at least) one semantic rule. For all English verbs and for Swedish main verbs, the verb-phrase rule above has a simple counterpart, but even for Swedish auxiliaries the treatment causes no problems, even though an extra case of the semantic rule had to be added in order to pass tense and aspect information properly, given that for main verbs, the tense information of the verb-phrase is the same as the one of the daughter verb and is simply unified up together with the other semantic information, while in the auxiliary case, the semantic interpretation of the mother verb-phrase still is the one of the daughter verb-phrase, but the tense is to be taken from the auxiliary.

Thus we get the following two (indeed very simplified!) semantic rules:

```
sem(vp_v_comp_Normal, mainv,
[(V,vp:[tense_and_aspect=TA],
  (V,v:[aux=n, tense_aspect=TA,
        subcat=Complements])
  | Complements
]).

sem(vp_v_comp_Normal, aux,
[(V,vp:[tense_and_aspect=TA],
  (Aux,v:[aux=y, tense_aspect=TA,
          subcat=(V,vp:[])]),
  (V,vp:[])
]).
```

Note that each constituent in the semantic rules is a pair with the first part holding the semantic logical-form fragment and the second part holding the (basically) syntactic information.

3.4 Negation

A specific case where the English and Swedish grammar differs significantly is in the treatment of negation. Negation in Swedish is expressed with the particle '*inte*' (not), which is placed after the main verb in a main clause, but before it in a subordinate clause, thus:

Han snarkade inte.	*He did not snore.*
...att han inte snarkade.	*...that he did not snore.*

Similar considerations also apply to a number of other common adverbials (so-called "mobile adverbs"), including '*ofta*' (often), '*alltid*' (always) and '*troligen*' (probably).

Even though negation tends to be used to a very small degree in the ATIS domain, a serious natural-language processing system must of course treat it, however, it does cause some problems both for English-Swedish transfer and for the QLF-formalism as such. The design choice in the English CLE was to treat negation semantically as an operator on the sentence structure which at the syntactic level pre-modifies a verb-phrase forming a new verb-phrase, the rule thus being schematically:

```
VP -> not VP
```

In Swedish such a treatment does not suffice; negation is still viewed as an operator at the semantic level, but instead of modifying verb-phrases, it is taken as modifying the verb itself in the syntax. Since whether the modification is pre- or post- depends on the type of clause, this has been treated by adding a **subordinate** feature to S, VP and V.

Three rules for verbs are needed, the first treating main clause negation, the second treating subordinate clause negation and the third treating a special case of main clause negation with a pronoun as object:

1. *mannen [gillade inte] Maria/mig* *the man did not like Mary/me*

```
v:[subordinate=n, ...]
 -->
v:[...]
 +
neg:[]
```

2. *att mannen [inte gillade] Maria/mig* *that the man did not like Mary/me*

```
v:[subordinate=y, ...]
 -->
neg:[]
 +
v:[vform=(\(att)), ...]
```

3. *mannen [gillade mig inte]*   *the man did not like me*

```
v:[subordinate=n, subcat=Rest, ...]
 -->
v:[subcat=[Pro|Rest], ...]
 +
Pro
 +
neg:[]
```

At the semantic level, treating negation as an operator causes some problems. Mainly since all mobile adverbs ought to be treated in the same way, but introducing QLF-operators for all of them would hardly be feasible. Thus negation is actually the only mobile adverb treated by the present version of the Swedish grammar. This problem and the fact that while modification of English verb phrases occurs external to the VP, Swedish modifiers are internal can be taken as an argument against having a VP node at all in Swedish, or as basis for introducing a $\overline{V}$ node. The above treatment goes a bit along the way of the second alternative.

4 Swedish grammar coverage

Without going into more details of the Swedish grammar, we should note that its coverage on the ATIS task was increased substantially during the project.

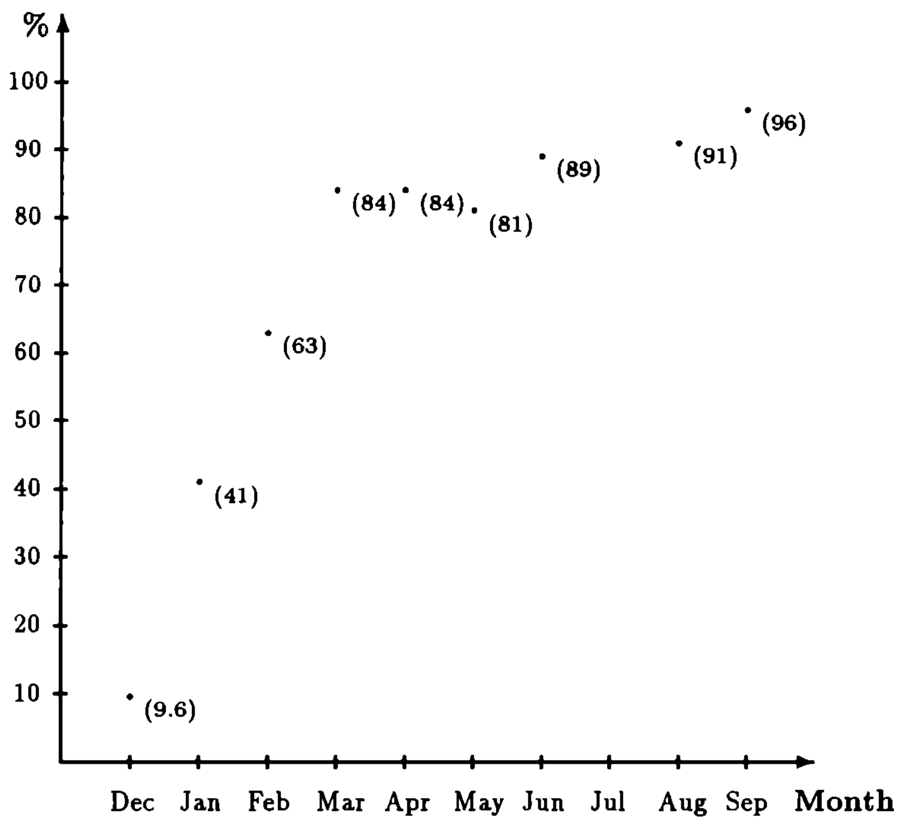

Figure 4: Transfer and generation coverage increase

Tests on a representative 281 sentence corpus showed an increase in coverage of the transfer and generation components combined from a mere 9.6% in mid-December 1992 to 96% in mid-September 1993, as can be seen in Figure 4.

As could be expected the main coverage increases were obtained early on in the project. After awhile, the coverage stabilized around 80%; to further increase the coverage, some major changes had to be undertaken, changes which at first actually lead to a slight coverage drop (as shown by the figure for mid-June).

Note that the figures in the graph refer to sentences that obtained a translation, *any* translation. For a discussion of the translation quality, see (Agnäs *et al.*, 1994).

5 Future Work and Conclusions

In the paper, the Swedish language processing component of the SLT English-to-Swedish spoken language translation system has been described. The main emphasis has been on the grammar and its coverage, but the other modules of the language processing part have also been described. The overall SLT system prototype and its coverage after the first year of the project has only been briefly discussed, while the paper has focused on the different modules of the Swedish processing component. These have been described mainly on a pro-example type level, showing the various rule formalisms at work.

At the date of writing, work has just begun on a second phase of the SLT project. We intend to reverse the system, so that translation of spoken Swedish into spoken English will be possible. Even though the main part of the work needed for that will be on producing a Swedish speech recognition system, the language processing components will be extended quite a lot at the same time. Partly because the Swedish part of the system has not been extensively tried for language processing as opposed to just generation for awhile, partly because the new version of SLT also will include extended processing in a new spoken language database query task, as well as allowing for some translations in a computer mediated person-to-person dialogue setup.

In parallel, work will be undertaken on systematically testing how the grammar coverage of the Swedish system can be tuned towards a new domain (Berglund and Gambäck, 1995) and whether the system is robust enough to be used as the basis for building a tree-bank of Swedish analyses (Santamarta *et al.*, 1995). Both these tests will use the representative Swedish "Stockholm-Umeå corpus" (SUC) (Ejerhed *et al.*, 1992).

6 Acknowledgements

The work reported here was funded by the Swedish Institute of Computer Science and Telia Networks.

I would like to thank Ivan Bretan, Jussi Karlgren and Christer Samuelsson who all were involved in the SLT project at SICS, my other colleagues in the SICS NLP-group, everybody else working on SLT related issues at the various sites, in particular Manny Rayner and Malgorzata Styś for some useful suggestions. Charlotta Berglund, Nikolaj Lindberg and Lena Santamarta are worthy of many thanks for (unwillingly) having had to help out in debugging the grammar while doing their BA and MA Thesis work.

References

Agnäs, M.-S., Alshawi, H., Bretan, I., Carter, D., Ceder, K., Collins, M., Crouch, R., Digalakis, V., Ekholm, B., Gambäck, B., Kaja, J., Karlgren, J., Lyberg, B., Price, P., Pulman, S., Rayner, M., Samuelsson, C., and Svensson, T. 1994. Spoken Language Translator: First-Year Report. Joint Research Report R94:03 and CRC-043, SICS and SRI International, Stockholm, Sweden and Cambridge, England.

Alshawi, H., editor. 1992. *The Core Language Engine.* The MIT Press, Cambridge, Massachusetts.

Alshawi, H. and Carter, D. 1994. Training and Scaling Preference Functions for Disambiguation. *Computational Linguistics*, 20:635–648.

Alshawi, H. and van Eijck, J. 1989. Logical Forms in the Core Language Engine. In *Proceedings of the 27th Annual Meeting of the Association for Computational Linguistics*, pages 25–32, Vancouver, British Columbia.

Alshawi, H., Carter, D. M., Gambäck, B., and Rayner, M. 1991. Translation by Quasi Logical Form Transfer. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics*, pages 161–168, University of California, Berkeley, California.

Bäckström, M., Ceder, K., and Lyberg, B. 1989. Prophon — An Interactive Environment for Text-to-Speech Conversion. In *Proceedings of the European Conference on Speech Communication and Technology*, pages 144–147, Paris, France.

Berglund, C. and Gambäck, B. 1995. On Testing Domain Adaptability. In *Proceedings of the 10th Scandinavian Conference on Computational Linguistics*, Helsinki University, Helsinki, Finland. (presentation).

Carter, D. 1995. Rapid Development of Morphological Descriptions for Full Language Processing Systems. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics*, pages 202–209, University College of Dublin, Dublin, Ireland.

Ejerhed, E., Källgren, G., Wennstedt, O., and Åström, M. 1992. The Linguistic Annotation System of the Stockholm–Umeå Corpus Project. Report 33, Department of General Linguistics, University of Umeå, Umeå, Sweden.

Gambäck, B. 1993a. On Implementing Swedish Tense and Aspect. In *Proceedings of the 9th Scandinavian Conference on Computational Linguistics*, pages 97–109, Stockholm University, Stockholm, Sweden.

Gambäck, B. 1993b. Towards a Uniform Treatment of Swedish Verb Syntax and Semantics. In *Proceedings of the 14th Scandinavian Conference of Linguistics and the 8th Conference of Nordic and General Linguistics*, pages 123–134, University of Gothenburg, Gothenburg, Sweden.

Gambäck, B. 1995. *Processing Swedish Sentences: A Unification-Based Swedish Grammar and some Applications for It.* Doctor of Engineering Thesis, Stockholm University/Royal Institute of Technology, Stockholm, Sweden.

Gambäck, B. and Bretan, I. 1994. Complex Verb Transfer Phenomena in the SLT System. In *Proceedings of the 1st Conference of the Association for Machine Translation in the Americas*, pages 89–96, Columbia, Maryland.

Gambäck, B. and Eineborg, M. 1995. A Grammar-Based Rule Formalism for a Text-to-Speech Interface System. In *Proceedings of the 5th Scandinavian Conference on Artificial Intelligence*, Trondheim, Norway. (to appear).

Gambäck, B. and Ljung, S. 1993. Question Answering in the Swedish Core Language Engine. In *Proceedings of the 4th Scandinavian Conference on Artificial Intelligence*, pages 212–225, Stockholm, Sweden. Also available as SICS Research Report, R92014, Stockholm, Sweden.

Gambäck, B. and Rayner, M. 1992. The Swedish Core Language Engine. In *Papers from the 3rd Nordic Conference on Text Comprehension in Man and Machine*, pages 71–85, Linköping University, Linköping, Sweden. Also available as SICS Research Report, R92013, Stockholm, Sweden.

Kiefer, F. 1970. *Swedish Morphology*. Skriptor, Stockholm, Sweden.

Koskenniemi, K. 1983. *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production*. Doctor of Philosophy Thesis, University of Helsinki, Helsinki, Finland.

Murveit, H., Butzberger, J., and Weintraub, M. 1991. Speech Recognition in SRI's Resource Management and ATIS Systems. In *Proceedings of the 4th Speech and Natural Language Workshop*. DARPA, Morgan Kaufmann.

Parkinson, S. 1992. A Computational Grammar for Use in Machine Translation. Master of Philosophy Thesis, Cambridge University, Cambridge, England.

Rayner, M. and Bouillon, P. 1995. Hybrid Transfer in an English–French Spoken Language Translator. (In manuscript.).

Rayner, M., Alshawi, H., Bretan, I., Carter, D. M., Digalakis, V., Gambäck, B., Kaja, J., Karlgren, J., Lyberg, B., Pulman, S. G., Price, P., and Samuelsson, C. 1993. A Speech to Speech Translation System Built from Standard Components. In *Proceedings of the Workshop on Human Language Technology*, Princeton, New Jersey. ARPA, Morgan Kaufmann.

Rayner, M., Carter, D. M., Price, P., and Lyberg, B. 1994. Estimating Performance of Pipelined Spoken Language Translation Systems. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, Kyoto, Japan.

Samuelsson, C. 1994. Notes on LR Parser Design. In *Proceedings of the 15th International Conference on Computational Linguistics*, volume 1, pages 386–390, Kyoto, Japan.

Samuelsson, C. and Rayner, M. 1991. Quantitative Evaluation of Explanation-Based Learning as an Optimization Tool for a Large-Scale Natural Language System. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, pages 609–615, Sydney, Australia.

Santamarta, L., Lindberg, N., and Gambäck, B. 1995. Towards Building a Swedish Treebank. In *Proceedings of the 10th Scandinavian Conference on Computational Linguistics*, Helsinki University, Helsinki, Finland. (presentation).

Styś, M. 1995. Incorporating Discourse Aspects in English–Polish MT: Towards Robust Implementation. (In manuscript.).